

**Vysoká škola báňská – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky**

**Návrh GUI v prostředí Matlab pro extrakci řečových
příznaků**

GUI design for features extraction in Matlab

2015

Jakub Jánošík

Zadání bakalářské práce

Student:

Jakub Jánošík

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2601R013 Telekomunikační technika

Téma:

Návrh GUI v prostředí Matlab pro extrakci řečových příznaků
GUI Design for Features Extraction in Matlab

Zásady pro vypracování:

Cílem práce je vytvoření grafického uživatelského rozhraní v prostředí Matlab pro extrakci řečových příznaků pomocí openSMILE.

1. Popis vzniku a vlastností řeči.
2. Popis řečových příznaků a použitého softwaru.
3. Návrh GUI v prostředí Matlab pro extrakci řečových příznaků z projektu openSMILE.
4. Ověření funkčnosti navrženého GUI.

Seznam doporučené odborné literatury:

ZAPLATÍLEK, Karel a Bohuslav DOŇAR. MATLAB: tvorba uživatelských aplikací. 1. vyd. Praha: BEN, 2004, 215 s. ISBN 80-730-0133-0.


PSUTKA, Josef, Luděk MULLER, Jindřich MATOUŠEK a Vlasta RADOVÁ. Mluvíme s počítačem česky. Vyd. 1. Praha: Academia, 2006, 746 s. ISBN 80-200-1309-1.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Jaromír Továrek**


Datum zadání: 01.09.2014

Datum odevzdání: 07.05.2015



doc. Ing. Miroslav Vozňák, Ph.D.
vedoucí katedry



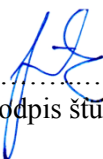


prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prehlásenie študenta

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

V Ostrave dne: 4. mája 2015


.....
podpis študenta

Pod'akovanie

Rád by som poďakoval *Ing. Jaromírovi Továrekovi* za odbornú pomoc, cenné rady, znalosti a konzultáciu pri vytváraní tejto bakalárskej práce.

Abstrakt

Cieľom tejto práce je navrhnuť GUI (Graphical User Interface) v programovom prostredí Matlabu pre extrakciu rečových príznakov. GUI bude navrhnuté pre uľahčenie práce s programom OpenSmile.

Prvá časť práce oboznamuje s teoretickým základom vzniku a vlastností reči, parametrizáciu rečového signálu, ktorá je veľmi dôležitá pre pochopenie výpočtu programu OpenSmile.

Druhá časť práce zahŕňa krátky popis použitých programov Matlabu a OpenSmile a zostavením samotného GUI.

Kľúčové slová

GUI; button; LPC; Matlab; MFCC; OpenSmile

Abstract

The aim of this work is to design a GUI (Graphical User Interface) in MATLAB programming for extracting speech features. GUI is designed to facilitate working with the program OpenSmile.

The first part introduces the theoretical basis for the creation and characteristics of speech parameterization of the speech signal, which is very important for understanding the calculation program OpenSmile.

The second part includes a brief description of the program Matlab and OpenSmile and establishing itself GUI.

Key words

GUI; button; LPC; Matlab; MFCC; OpenSmile

Zoznam použitých symbolov

Symbol	Jednotky	Význam symbolu
E	—	Energia sila hlasu
F_0	Hz	Základná frekvencia reči
G	—	Zosilnenie zdroja budenia
$H(z)$	—	Prenosová funkcia
l_{ram}	—	Dĺžka rámca
N	—	Dĺžka rečového signálu
N_{ram}	—	Celkový počet rámcov
p_{ram}	—	Veľkosť prekrytia
$s[n]$	—	Pôvodná hodnota signálu
s_{ram}	—	Posun rámca
T_0	s	Základná perióda reči
$w(n)$	—	Hammingová oknová funkcia
$x[n]$	—	Rečový signál
μ_s	—	Stredná hodnota signálu

Obsah

Úvod	- 9 -
1 Vznik a vlastnosti reči	- 10 -
1.1 Proces vytvárania reči človekom	- 10 -
1.2 Model procesu vytvárania reči	- 12 -
1.3 České rečové jednotky	- 14 -
2 Parametrizácia rečového signálu	- 15 -
2.1 Základné parametre rečového signálu	- 18 -
2.2 Kepstrum	- 20 -
2.2.1 Kepstrálne koeficienty LPC	- 21 -
2.2.2 Kepstrálne koeficienty MFCC	- 22 -
2.2.2.1 Postup výpočtu MFCC	- 22 -
3 Použité programy	- 25 -
3.1 Matlab	- 25 -
3.1.1 GUIDE - nástroj pre interaktívnu tvorbu grafického rozhrania	- 25 -
3.2 OpenSmile	- 27 -
3.2.1 Architektúra OpenSmile	- 28 -
4 Návrh GUI pre extrakciu rečových príznakov	- 29 -
Záver	- 39 -
Použitá literatúra	- 40 -
Zoznam príloh	- 41 -

Úvod

Základný a najprirodzenejším spôsob prenosu informácie medzi ľuďmi je prostredníctvom hovorovej reči. V súčasnej dobe sa vedci snažia, aby sa mohol i počítač zapojiť do dialógu s človekom. Komunikácia s počítačom a človekom je v súčasnej modernej dobe veľmi prospešná a často môže človeku uľahčiť život. Ak v súčasnej dobe chceme, aby sa počítač stal plnohodnotným partnerom človeka v dialógu, musíme vyriešiť niekoľko komplikovaných úloh, medzi tieto úlohy patrí spracovanie rečového signálu, počítačovej syntézy a automatického rozpoznávanie reči. V súčasnej dobe je ešte stále nedostupný plnohodnotný dialóg človeka s počítačom bez akéhokoľvek obmedzenia.

Cieľom mojej bakalárskej práce je navrhnuť GUI v programe Matlab pre extrakciu rečových príznakov, pre uľahčenie výpočtov v programu OpenSmile. V prvej časti mojej práce popisujem vznik a vlastnosti reči. V tejto časti popisujem ako prebieha a vzniká základný tón reči. V druhej časti popisujem parametrizáciu rečového signálu. Táto časť popisuje metódy predspracovania rečové signálu. Taktiež popisuje metódy výpočtu, ktoré spracováva program OpenSmile ako sú parametre LPC, MFCC a prozodiu. V tretej časti popisujem programy, s ktorými som pracoval. Poslednou časťou mojej práce je návrh a realizácia môjho navrhnutého GUI. V tejto časti popisujem založenie GUI od základnou až po konečnú funkčnosť.

1 Vznik a vlastnosti reči

1.1 Proces vytvárania reči človekom

Na to aby človek mohol hovoriť, využíva svoje rečové orgány(artikulačné). Rečové orgány sa skladajú z hlasiviek, nosnej ústnej a hrdelnej trubice, mäkkého a tvrdého podnebia, zubov a jazyka. Základná funkcia týchto rečových orgánov v našom tele často spolu ani nesúvisí(dýchanie, prijímanie potravy atď.) Spája ich až tvorba reči. Z hľadiska tvorby reči ich tvorí hlasový trakt. U dospelého muža je celková dĺžka rečových orgánov až 17 cm a plocha pričného prierezu až asi okolo 20 cm². Hlasový trakt človeka môžeme rozdeliť do troch základných skupín a to na dýchacie ústrojenstvo, hlasové ústrojenstvo, artikulačné ústrojenstvo (obrázok 1.1). [3]

Dýchacie ústrojenstvo- Slúži primárne k základným rečovým funkciám (dýchaniu). Pri reči sa fyziologické dýchanie mení. Zatiaľ čo pomer vdychu a výdychu je asi 2:3(16 až 20 krát za minútu) pri reči sa však nápadne skraca, vdychom a výdychom sa predlžuje(1:7 až 1:12). Maximálny rozdiel medzi úplne naplnenými pľúcami a úplne prázdnyimi je asi 5 litrov. energiu pre reč získavame pri nádychu. Hovorený prejav sa realizuje pri výdychu, ktorý predĺži a zaisť trvalý tlak v hrtane, čo je dôležité pri vzniku hlasu. Prúd vzduchu je z pľúc odvedený priedušnicou (tracheou). Potom prechádza hrtanom a nad hrtanovými dutinami, kde sa modifikuje a ako rečový signál je vyžarovaný perami do priestoru.[3]

Hlasové ústrojenstvo - Hlasové ústrojenstvo je uložené v hrtane. Hrtan je s pľúcami spojený s priedušnicou. Najdôležitejšiu časť tvoria hlasivky. Nachádzajú sa v hrtanovej dutine priamo za ohryzkom. V dôsledku kmitania hlasiviek sa vzduchový prúd rozdelí a tak vznikne vzduchová vlna, ktorú my vnímame ako vzduch. Tento periodický prúd vzduchových pulzov tvorí základný tón ľudského hlasu a predstavuje nosný zvuk reči. Základný hlasový tón má svoju frekvenciu, ktorú označujeme ako F_0 a nazývame ju fundamentálna frekvencia. Je fyzikálnou charakteristikou rečového signálu a odpovedá výške hlasu. Jej prevrátenú hodnotu $T_0=1/F_0$ nazývame perióda základného hlasivkového tónu a udáva že tón má rozsah asi 60-400 Hz.[3]

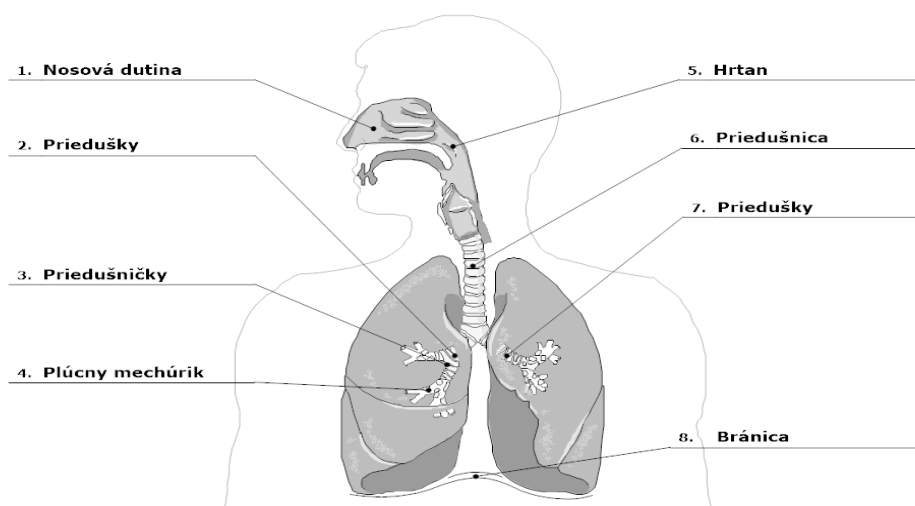
Artikulačné ústrojenstvo - Je to posledné ústrojenstvo, ktoré sa podieľa na tvorbe reči a je uložené nad hrtanom. Skladá sa z troch dutín a to hrdlová dutina, nosná dutina a ústna dutina. Medzi najvýznamnejšie artikulatory patrí jazyk, pery a mäkké podnebie a podieľajú sa na vytváraní najväčšieho počtu zvukov. Medzi ďalšie artikulatory patria aj zuby, tvrdé podnebie alebo čeľusť a hrtan. Najdôležitejší zo všetkého je jazyk, veľmi pružný a prispôsobivý.

Je schopný tvoriť veľa tvarou a dokáže rýchlo prechádzať z jednej pozície do druhej. Variabilita jazyka vytvára mnoho tvarou ústnej dutiny a tým zaisťuje vytváranie rôznych zvukov reči.[3]

V nad hrtanových dutinách dostáva zvuk výrazné zmeny, kam ako výdychový prúd vzduchu postupuje z hlasového ústrojenstva. Tieto zmeny majú dva druhy:

Vytváranie tónovej štruktúry - ku zmene rozloženia akustickej energie vo vznikajúcom rečovom signáli dochádza vplyvom rezonancie vo vnútri ústnej, nosnej a hrdlovej dutiny. Táto akustická energia sa sústreďuje okolo formantových frekvencií. Formanty sú oblasti zosilnenia akustickej energie a značíme ich číslami. Pokiaľ sa do vytvárania reči zapojí aj nosná dutina, dochádza k antiformantu, to je potlačenie niektorých frekvenčných oblastí. Vzniká zložený zvuk tónového charakteru, čiže hlas. Tvorí podstatu predovšetkým samohlások, ktoré sú znelé časti reči. [3]

Vytváranie šumovej zložky - priechod vzduchu je ovplyvnený nadhrtanovými dutinami. Na rôznych miestach môžu svojím pohybom zúžiť priechod, úplne ho uzavrieť alebo uvoľniť. Potom sa vytvoria rôzne prekážky, cez ktoré sa prediera vydýchnutý vzduch a podľa charakteru prekážky vzniká rôzny šum. Šum tvorí základ spoluhlások.[3]



Obrázok 1.1: Orgány človeka pri vzniku reči

1.2 Model procesu vytvárania reči

Pri vytváraní reči počítačom je dobré si najskôr určiť modelový proces vytvárania reči a potom sa riadiť podľa tohto modelu. Z výpočtového hľadiska by bolo dobré, aby takýto model bol lineárny a časovo invariantný. Ľudská reč je súvislý časovo premenný proces, preto doteraz nebol predložený žiadny univerzálny model reči, ktorý by spĺňal požiadavky fyziológie, akustiky či špecialistami na spracovanie signálu.

Model hlasového traktu. Parametre modelu závisia na tvaru, prierezu a dĺžke nosnej, hrdlovej a ústnej dutiny. Sú rozdielne pre každého človeka, ktorý vyslovuje ten istý zvuk. Významne sa menia aj pri vyslovovaní iných zvukov, čo je dané objemom dutiny, polohou jazyka apod. Prenosovú funkciu celo polového modelu hlasového traktu $V(z)$ ide zapísať v tvare (1.1) [3]

$$V(z) = \frac{1}{\prod_{i=1}^k [1 - 2 \exp(-\alpha_i T) \cos(\beta_i T) z^{-1} + \exp(-2\alpha_i T) z^{-2}]} \quad (1.1)$$

Kde k je počet formantov, ktoré model rešpektuje, α_i a β_i su neznáme koeficienty modelu a T je perióda vzorkovania.

Model vyžarovania zvuku. Fyzikálne javy pre akustické riadiace reči matematicky popísal George Fant. Ktorý odvodil prenosovú funkciu modelu vyžarovania zvuku do tvaru

$$L(z) = 1 - z^{-1} \quad (1.2)$$

Budenie modelu rečovej produkcie. Pre neznelé zvuky je vstupom rečovej produkcie náhodný šum a pre znelé zvuky je vstupom sled pulzov s periódou T_0 . Pri perióde vzorkovania T je budila funkcia v diskkrétnej oblasti nenulová v časových okamžikoch, tie odpovedajú perióde základného hlasivkového tónu T_0 , to znamená $T_0 = bT$. Koeficient b znázorňuje koľkokrát je prima základného tónu dlhšia než vzorkovanie. Ak položíme periódu vzorkovania $T=1$, môžeme ju vyjadriť do vzťahu (1.4). [3]

$$U(v) = G \sum_{n=0}^{\infty} (z^{-b})^n = \frac{G}{1 - z^{-b}} \quad (1.3)$$

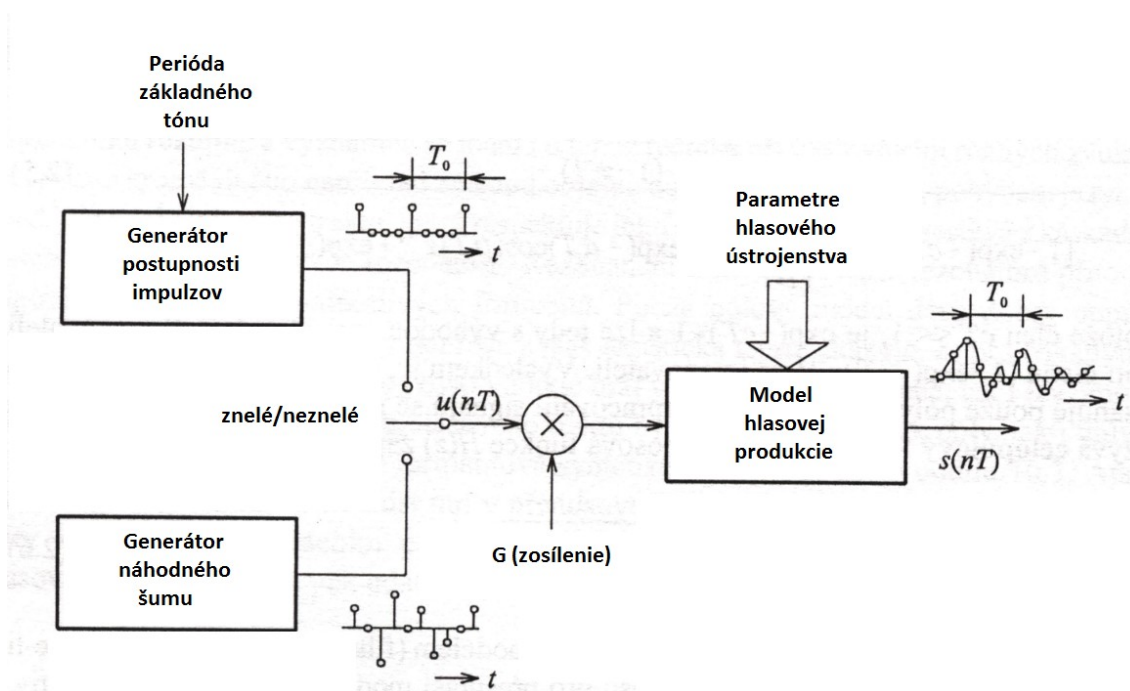
pričom G nazývame zosilnenie zdroja budenia. Pre diskkrétne modely môže byť zdroj náhodného zvuku šumu realizovaný generátorom náhodných čísel.

Celkový model produkcie reči. Pre prípad celkového modelu reči je potrebné zjednotiť submodely hlasového traktu, hlasiviek a radiace do modelu s prenosovou funkciou $H(z)$

$$H(z) = G(z) V(z) L(z) \quad (1.4)$$

Člen $cT \ll 1$, tak je $\exp(-cT) \approx 1$ a môžeme previesť úpravu vyrušením jedného člena v čitateli s členom v menovateli $[1 - \exp(-cT)z^{-1}]$. Výsledkom tejto produkcie bude model reči obsahovať dva póly, čo sa v terminológii spracovania signál nazýva **celo pólový filter**.

Ak zoberieme do úvahy všetky doposiaľ uvedené skutočnosti, tak môžeme model reči popísať z obrázka 1.2. Parametre tohto modelu sú premenné v čase, ale na konkrétnom časovom interval (10-30 ms) sú brané ako konštantné parameter, takého parameter nazývame **krátkodobé modely**. Oba zdroje budenia treba určiť nezávisle na zosilnení a parameter G pripojiť ako parameter v čase sa pomaly meniaci. [3]



Obrázok 1.2: Model vytvárania reči

Uvedený model sa využíva pre získanie širokého spektra reprezentácie rečového signálu. Toto spektrum môžeme zoradiť podľa zložitosti od jednoduchého vzorkovania rečových kmitov (kódovanie tvaru vlny) až po odhad parametru modelu.[3]

1.3 České řečové jednotky

České řečové jednotky můžeme rozdeliť do dvoch hlavných skupín na **samohlásky** a **spoluhlásky**. Znalosti akustických a artikulačných vlastností řečových jednotek využívají do značné míry aj metody počítačové syntézy řeči. V metodách automatického spracovania řečového signálu můžeme tiež využit znalosti akustických charakteristik českých jednotek (například při návrhu vhodné parametrizace řeči). [3]

Samohlásky - V Českém jazyce rozlišujeme 10 samohlásek: /a/, /e/, /i/, /o/, /u/, /á/, /é/, /í/, /ó/, /ú/. Při jejich artikulaci se liší od sebe nastavením hlasového traktu(/a/-/e/, /i/-/o/) a délkou(/a/-/á/, /e/-/é/, /i/-/í/, /o/-/ó/, /u/-/ú/). Krátke a dlhé samohlásky se odlišují především trváním. Samohlásky jsou nositelkami tónové kvality a estetiky hovorové řeči. Samohlásky lze snadno rozpoznat v řečovém signálu díky vyšší amplitudě a kvaziperiodickému průběhu (s periodou základního tónu). V Českém jazyce mají zvláště postavení tři **dvojhlásky** /o u/, /a u/, /e u/, přičemž druhé a třetí jsou příznakem cizosti slova. Jedinou pravou dvojhláskou je /o u/. [3]

Spoluhlásky - Patří do druhé skupiny fonémů. Český jazyk má 27 spoluhláskových fonémů a to /p/, /b/, /t/, /d/, /tʃ/, /dʃ/, /k/, /g/, /f/, /v/, /s/, /z/, /š/, /ž/, /ch/, /h/, /c/, /dz/, /č/, /ř/, /m/, /n/, /ň/, /l/, /r/, /j/. Spoluhlásky patří k základní opoře zrozumitelnosti řeči. Při porovnání akustického signálu samohlásek a spoluhlásek, spoluhlásky mají příznačný šum a menší amplitudu. [3]

2 Parametrizácia rečového signálu

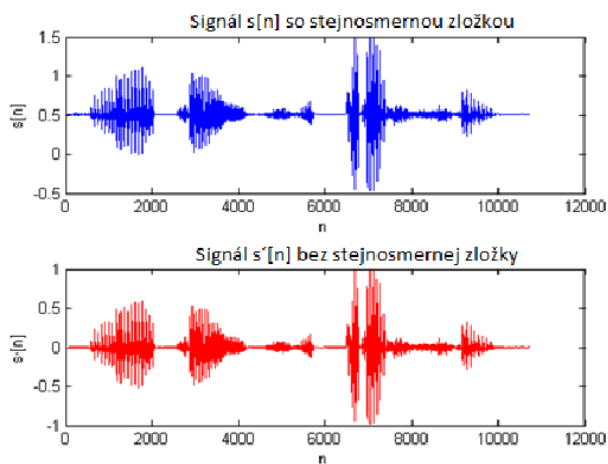
Vstupný blok predspracovania rečového signálu z jeho akustickej formy na súbor vektorov parametrov je spoločný znak systémov rozpoznávania reči a nazývame ho parametrizácia reči (extrakcie príznakov). Metódy parametrizácie reči sú založené na myšlienke nájdenia vhodného modelu rečového signálu. Pri parametrizácii sa vyberú len tie dáta, ktoré sú potrebné pre ďalšie spracovanie. Parametre môžeme rozdeliť na **skalárne** (úsek reči je reprezentovaný len jedným číslom, medzi tieto parametre patrí energia signálu, počet prechodov nulou, základný tón reči atď.) A **vektorové** (úsek reči je reprezentovaný vektorom hodnôt, medzi vektorové parametre patrí MFCC, LPC atď.).[3]



Obrázok 2.1: *Predspracovanie rečového signálu*

Odstránenie jednosmernej zložky - Rečový signál môže obsahovať jednosmernú zložku, ktorá však nenesie žiadnu užitočnú informáciu, na druhej strane môže byť však rušivá pri určovaní niektorých charakteristík rečového signálu (počet prechodu nulou, energií signálu). Pokiaľ signál nie je spracovaný v reálnom čase, tak jednosmernú zložku odstránime odčítaním strednej hodnoty signálu μ_s od pôvodného signálu $s[n]$ (2.1).[3]

$$s'[n] = s[n] - \mu_s \quad (2.1)$$

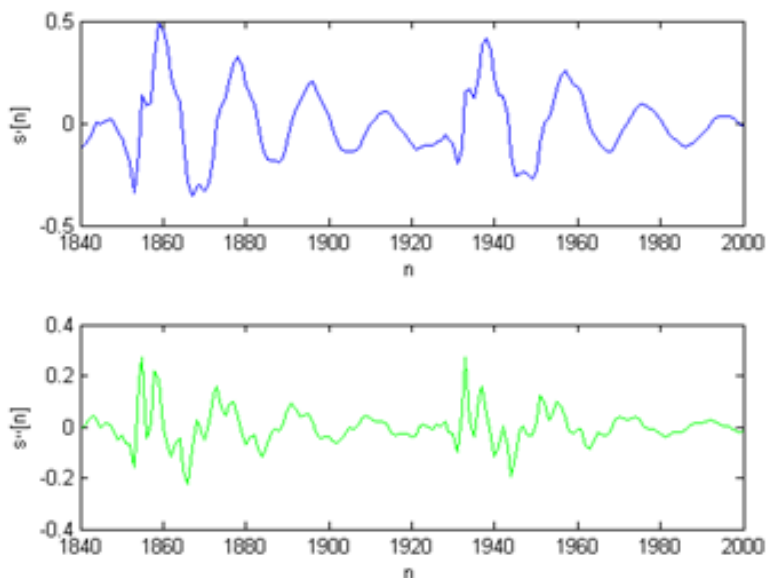


Obrázok 2.2: *Priebeh signálu $s[n]$ as $s'[n]$*

Preemfáza - Jednou zo základných úprav rečového signálu pred jeho ďalším spracovaním je číslicová filtrácia tzv. preemfázovým filtrom, ktorý ma za úlohu zdôrazniť vyššie frekvenčné zložky rečového signálu, čím sa vyrovná jeho prirodzený útlm, spôsobený vyžarovacou charakteristikou úst a zároveň sa zdôraznia najvýznamnejšie zložky spektra reči. Realizovanie býva pomocou FIR filtru prvého rádu, ktorý má prenosovú funkciu $H(z)$:

$$H(z) = 1 - kz^{-1} \quad (2.2)$$

Parametre k volíme v rozsahu $k \in [0,91]$



Obrázok 2.3: *Priebeh segmentu signálu $s'[n]$ a $s''[n]$*

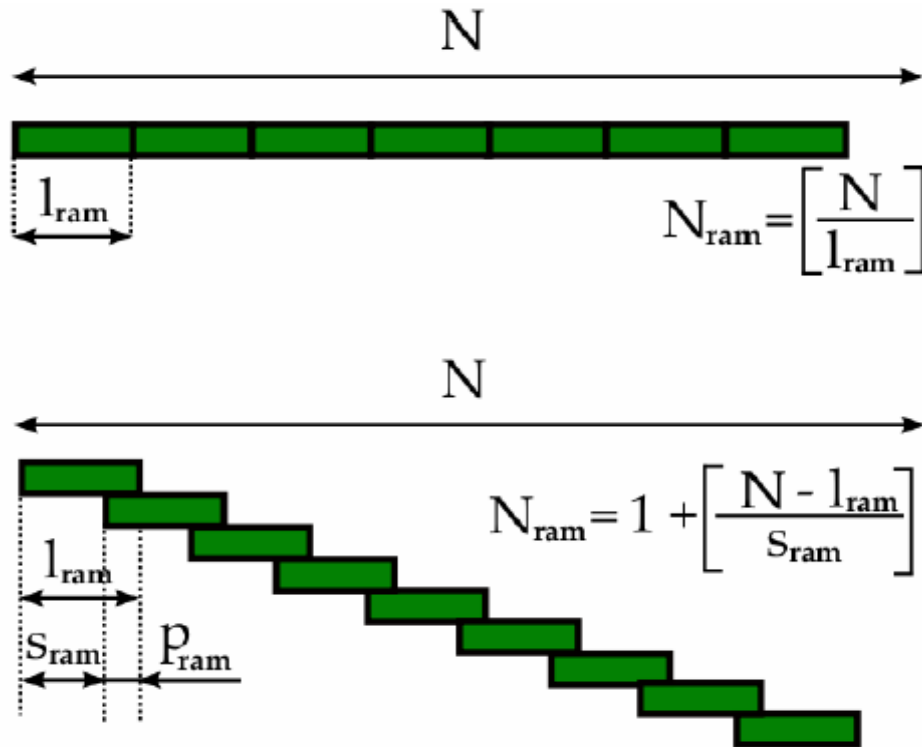
Segmentácia - Pre zisťovanie parametrov rečového signálu je potrebné, aby bol signál stacionárny, pretože rečový signál nie je stacionárny je potrebné tento signál rozdeliť na menšie úseky (segmenty), kde bude signál stacionárny. Na základe zotrvačnosti hlasového traktu volíme dĺžku týchto rámcov l_{ram} . Aby sme mohli predpokladať, že je signál stacionárny, aby si rámec zachoval periódu musí byť dostatočne malá a zároveň dostatočne dlhá. Dĺžka rámca sa spravidla volí v rozmedzí 20 - 30ms, čo pri vzorkovacej frekvencii $F_v = 16\text{kHz}$ predstavuje 320 - 480 vzoriek. Rámec je vhodné navzájom prekryvať, pretože sa parametre signálu v susedných rámcov značne líšia. Zvolíme vhodnú veľkosť prekrytia p_{ram} , obvykle sa volí prekrytie rámca 10-15 ms. Posun rámca potom vypočítame pomocou vzťahu (2.3). [3]

$$s_{ram} = l_{ram} - p_{ram} \quad (2.3)$$

Celkový počet rámcov príslušného rečového signálu vypočítame podľa vzťahu (2.4), ktorý je zobrazený aj na obrázku (2.4).

$$N_{ram} = 1 + \left\lceil \frac{N - l_{ram}}{s_{ram}} \right\rceil \quad (2.4)$$

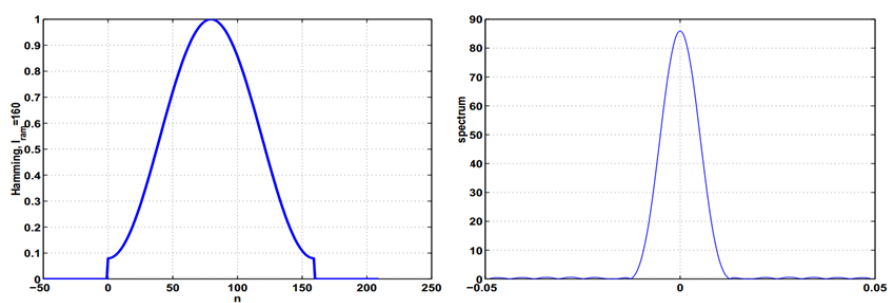
kde N_{ram} je celkový počet rámcov, N je dĺžka rečového signálu, l_{ram} je dĺžka rámca vo vzorcoch a s_{ram} je posun rámca vo vzorcoch.



Obrázok 2.4: Dĺžka signálu na rámce

Váhovanie okien - Oknová funkcia potláča váhy jednotlivých vzoriek na začiatku a na konci rámca. Váhovanie zmierňuje medzi okrajmi jednotlivých rámcov, ktoré do nich vnikajú pri segmentácii rečového signálu. To je nežiaduce hlavne vo frekvenčnej analýze. Zamedzenie tohto prechodu sa prevádza vynásobením každého rámca oknovou funkciou. V oblasti spracovania reči sa používa Hammingová oknová funkcia $w(n)$, ktorú vyjadríme vzťahom (2.5).

$$w(n) = 0,54 - 0,46 \cos \frac{2\pi n}{l_{ram}}, \text{ pre } 0 \leq n \leq l_{ram} \quad (2.5)$$


 Obrázok 2.5: *Hammingovo okno v časovej a frekvenčnej oblasti*

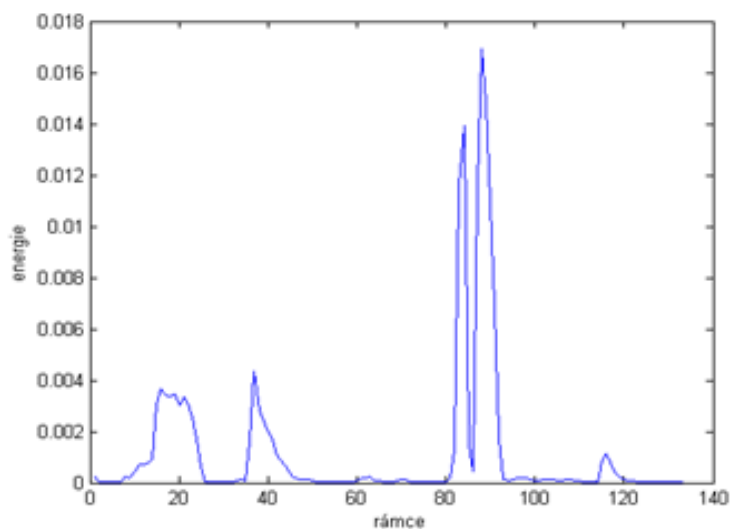
2.1 Základné parametre rečového signálu

Energia - energiu si možno predstaviť ako silu hlasu (intenzitu). Energiu môžeme tiež využiť ako jednoduchý detektor rečovej aktivity alebo pre rozlíšenie znelých hlások (vysoká energia) od neznelých (nízka energia).

$$E = \frac{1}{l_{ram}} \sum_{n=1}^{l_{ram}} x^2[n] \quad (2.6)$$

l_{ram} – dĺžka rámca

$x[n]$ – rečový signál

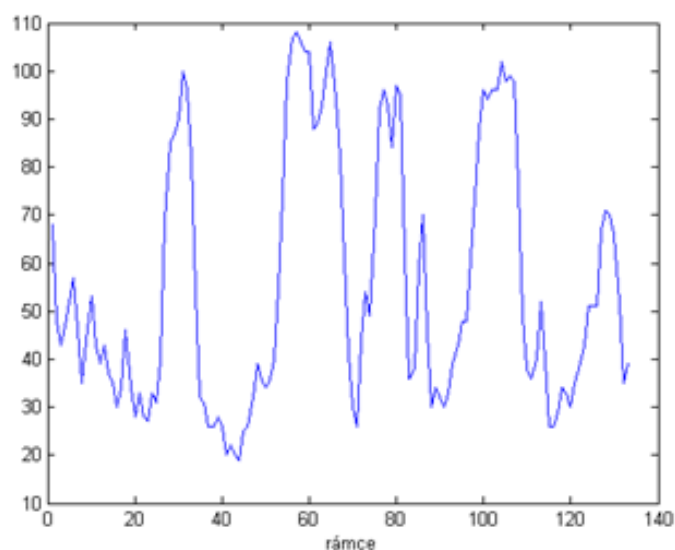

 Obrázok 2.6: *Priebeh energie signálu*

Počet priechodov nulou - zaznamenáva zmeny polarity signálu, počet prechodov nulou môžeme opäť využiť pre rozlíšenie hlások na znelé (malý počet prechodov) a neznelé (veľký počet prechodov nulou - podobné šumu). Na výpočet sa používa znamienková funkcia, ktorá nadobúda hodnoty 1 0 -1 podľa toho aká je hodnota signálu. Ak je signál v oboch susedných hodnotách rovnaký (kladný alebo záporný), tak je výsledok funkcie 0. Parameter počet priechodov nulou je veľmi citlivý na šum.[3]

$$ZCR = \sum_{n=1}^{l_{ram}} |\text{sign } x[n] - \text{sign } x[n-1]| \quad (2.7)$$

l_{ram} – dĺžka rámca

$x[n]$ – rečový signál



Obrázok 2.7: *Priebeh počtu priechodov nulou*

Základná frekvencia - predstavuje frekvenciu kmitania hlasiviek. Na základnej frekvencii má vplyv mnoho parametrov ako sú napríklad vek, pohlavie alebo emócie. Základnej frekvencii môžeme získať ako vo frekvenčnej oblasti tak v časovej alebo pomocou kepsrálnych koeficientov.

$$F_0 = \frac{1}{T_0} \quad [Hz] \quad (2.8)$$

F_0 - základná frekvencia reči

T_0 - základná perióda reči

Typické hodnoty základného tónu sú (muži 90-120 Hz, ženy 150-300 Hz, deti 350-400 Hz).

Jitter - Predstavuje malé kolísanie dĺžky základnej periódy a je veľmi závislý na emocionálnom stave hovoriaceho človeka. Definovať ho môžeme ako stredný rozdiel dĺžok susedných periód, delený strednou dĺžkou periódy a obvykle ho vyjadrujeme v percentách.

$$Jitter_{Rel} = \frac{\frac{1}{N-1} \sum_{i=1}^{N-1} |T_i - T_{i+1}|}{\frac{1}{N} \sum_{i=1}^N T_i} \quad (2.9)$$

Shimmer - Predstavuje kolísanie amplitúdy hlasivkových pulzov. Je definovaný stredným rozdielom medzi amplitúdou susedných hlasivkových pulzov a vyjadrujeme ho v decibeloch. Môžeme ho vypočítať pomocou vzťahu (2.10).

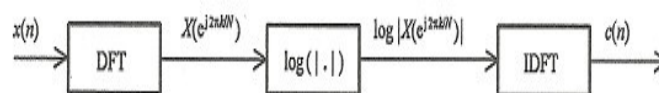
$$Shimmer(dB) = \frac{1}{N-1} \sum_{i=1}^{N-1} \left| 20 \log \left(\frac{A_{i+1}}{A_i} \right) \right| \quad (2.10)$$

2.2 Kepstrum

Kepstrum môžeme definovať ako spektrum spektra, ktoré vzniká procesom kepstrálnej analýzy. Cieľom kepstrálnej analýzy je vydeliť jednotlivé členy konvolutorného súčinu. Časť výsledného kepstra potom nesie informáciu o budiacej funkcii (základná ton reči) a druhá časť o impulzívnej odozve hlasového ústrojenstva. Reálne kepstrum potom môžeme vypočítať pomocou vzťahu (2.11).

$$c(n) = \operatorname{Re} \left\{ \frac{1}{N} \sum_{k=0}^{N-1} \ln |X(k) e^{j2\pi kn/N}| \right\}, n = 0, \dots, N-1, \quad (2.11)$$

kde N je počet vzorcov (dĺžka) mikrosegmentu. Postup výpočtov reálneho kepstra je zobrazený na obrázku 2.3, kde $x(n)$ je diskretný signál, $e^{j2\pi kn/N}$ je spektrum signálu $x(n)$, $\log |X(e^{j2\pi kn/N})|$ je prirodzený logaritmus modulu spektrálnej funkcie $x(n)$ a $c(n)$ je označené reálnym kepstrom $x(n)$.



Obrázok 2.8: Postup výpočtu reálneho kepstra

Kepstrálne koeficienty môžeme využiť v mnohých oblastiach spracovania reči (napr. určenie základnej frekvencie a pod.). V súčasnej dobe sú v systémoch rozpoznávania hovorovej reči preferované dve modifikácie homomorfneho spracovania - kepstrálne koeficienty LPC a kepstrálne koeficienty MFCC. [3]

2.2.1 Kepstrálne koeficienty LPC

Kepstrálne koeficienty LPC sú menej korelované ako napríklad koeficienty LPC, čo má výhodu v podpore ich využitia v systémoch rozpoznávania reči. Pri výpočte kepstrálnych koeficientov postupujeme v súlade s pravidlami homomorfného spracovania reči, teda vychádzame z prirodzeného logaritmu funkcie $H(z)$, to je $\ln(H(z)) = \ln\left(\frac{G}{A(z)}\right)$, kde $A(z)$ označujeme polynom Premennej z^{-1} Q -tého rádu. Aby sme mohli vypočítať kepstrálne koeficienty potrebujeme tieto vzťahy (2.12), (2.13), (2.14). [3]

$$c(1) = -a_1 \quad (2.12)$$

$$c(k) = -a_k - \sum_{i=1}^{k-1} \left(\frac{i}{k}\right) c(i)a_{k-i}, \text{ pre } 2 \leq k \leq Q, \quad (2.13)$$

$$c(k) = \sum_{i=1}^Q \left(\frac{k-i}{k}\right) c(k-i)a_i, \text{ pre } k > Q \quad (2.14)$$

Keďže kepstrálne koeficienty LPC s vyššími indexmi nadobúdajú štandardne nižších hodnôt, je z praktických dôvodov užitočné vykonať tzv. Liftering podľa vzťahu (2.15).

$$c_{lift}(n) = \left[1 + \frac{L}{2} \sin\left(\frac{\pi n}{L}\right)\right] c(n) \quad (2.15)$$

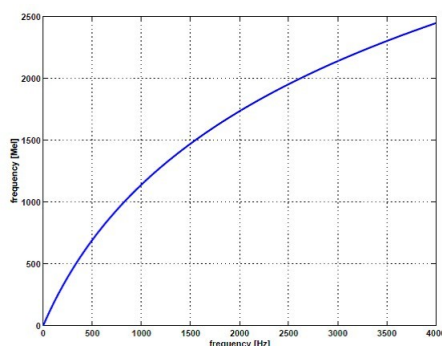
kde L je váha lifteringu s typickou hodnotou $L = 22$. Liftering získa parametre s približne podobnými hodnotami.

2.2.2 Kepstrálne koeficienty MFCC

Pomocou Melovských kepstrálnych koeficientov spracovania rečových signálov prináša najväčšiu výhodu v tom, že do istej miery rešpektujeme vnímanie zvuku ľudským uchom nelineárnymi vlastnosťami (ľudské ucho má na nižších frekvenciách väčšie rozlíšenie ako na vyšších frekvenciách). Keď chceme čo najviac priblížiť kepstrum počutia, využívame túto metódu hlavne u rozpoznávania reči. Melovské kepstrálne koeficienty sa snaží kompenzovať nelineárne vnímanie frekvencií pomocou banky trojuholníkových pásmových filtrov s lineárnym rozložením frekvencií v tzv. Melovské frekvenčné škále. Táto škála je definovaná vzťahom(2.16).[3]

$$f_m = 2595 \log\left(1 + \frac{f}{700}\right), \quad (2.16)$$

kde f je frekvencia v lineárnej škále a f_m je odpovedajúca frekvencia v nelineárnej melovskej škále. [3]



Obrázok 2.9: *Nelineárny prevod Hertzov na Mely*

2.2.2.1 Postup výpočtu MFCC

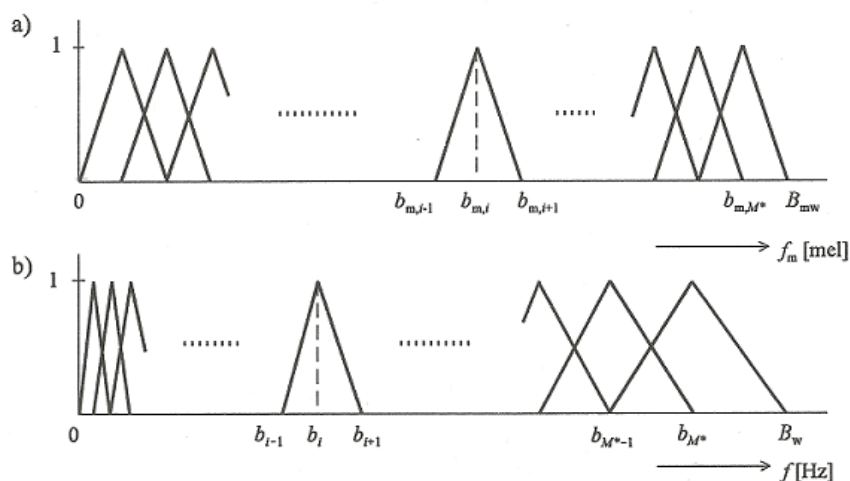
- **Vhodné predspracovanie rečového signálu a rozdelenie na mikrosegmenty**
- **Výpočet FFT analyzovaného signálu** - Výpočet amplitúdového spektra.

- **Melovská filtrácia** - Táto filtrácia sa realizuje pomocou banky trojuholníkových filtrov pozdĺž frekvenčnej osy s merátkom v melovskej škále. V závislosti na počte a umiestnenia kritických pásiem, je vhodné voliť počet pásiem tejto banky filtrov. Musíme ale rešpektovať celkovú šírku prenášaného pásma a veľkosť vzorkovacej frekvencie. Pre dané vzorkovacie frekvencie a šírky pásiem sú v tabuľke (2.1) sú uvedené najčastejšie používané počty pásiem (počty trojuholníkových filtrov v banke).Trojuholníkové filtre sú spravidla rozložené cez celé frekvenčné pásmo od nuly až po Nyquistovu frekvenciu.[3]

Tabuľka 2.1: *Typické hodnoty počtu filtrov M^* pre dané prenášané pásmo*

Frekvencia vzorkovania F_v [Hz]	8000	11000	16000	22000	44000
Prenášané pásmo $(0 ; B_w)$ [Hz]	(0;4000)	(0 ; 5500)	(0 ; 8000)	(0 ; 11000)	(0 ; 22000)
Prenášané pásmo $(0 ; B_{mw})$ [mel]	(0 ; 2146)	(0 ; 2458)	(0 ; 2840)	(0 ; 3174)	(0 ; 3921)
Počet pásiem M^*	15	17	20	22	27

Na obrázku 2.10 je uvedený príklad banky filtrov, kde má každý filter trojuholníkovú frekvenčnú odozvu, vzdialenosti filtrov a tiež šírka pásma sú určené konštantným intervalom v melovskej frekvenčnej škále. Priechod signálu filtrom teda znamená, že každý koeficient FFT je vynásobená zodpovedajúcim ziskom filtra a výsledky sú pre príslušné filtre akumulované.[3]

Obrázok 2.10: *Banka trojuholníkových filtrov*

a) melovská škála b) frekvenčná škála

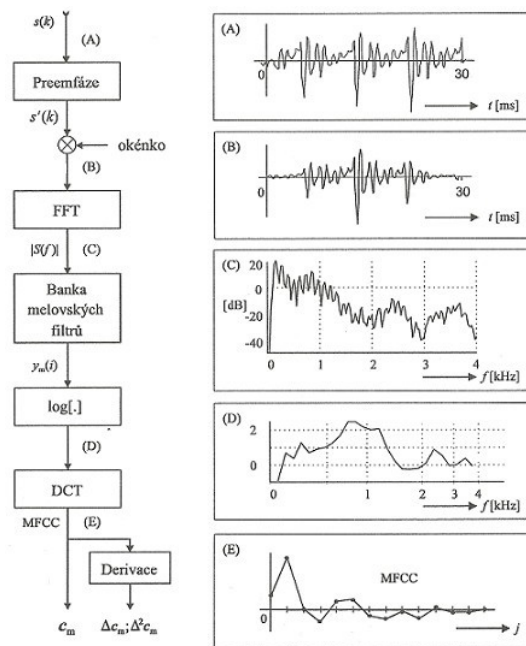
- **Výpočet logaritmov výstupu jednotlivých filtrov** - Logaritmovanie akumulovaných koeficientov sa vkladá do procesu spracovania dôležitých príznakov keprálnej analýzy, ktoré kladne obmedzujú dynamiku signálu.[3]

- **Výpočet spätnej diskkrétnej Fourierovej transformácie IDFT** - amplitúdové spektrum je reálne a symetrické a vzhľadom k tomu sa bude IDFT redukovať na diskretnú kosínusovú transformáciu DCT. Výstupom transformácie sú koeficienty MFCC (2.17).[3]

$$c_m(j) = \sum_{i=1}^{M^*} \log y_m(i) \cos\left(\frac{\pi j}{M^*}(i - 0,5)\right), \text{ pre } j = 0, 1, \dots, M, \quad (2.17)$$

kde $y_m(i)$ je odozva filtra, M^* je počet pásiem melovského pásmového filtra a M je počet melovských keprálnych koeficientov. Počet týchto koeficientov je možné voliť podstatne menší, než je počet melovského pásmového filtra M^* , obvykle stačí uvažovať $M=10-13$ koeficientov.[3]

- **Výpočet Delta a Delta-Delta (akceleračných) koeficientov** - Ku statickým koeficientom MFCC sa dopočítajú dynamické koeficienty (1. a 2. derivácie). Postup výpočtu melovských keprálnych koeficientov spolu s priebehmi signálov v jednotlivých bodoch je zobrazený na obrázku (2.11).



Obrázok 2.11: Postup výpočtu melovských koeficientov

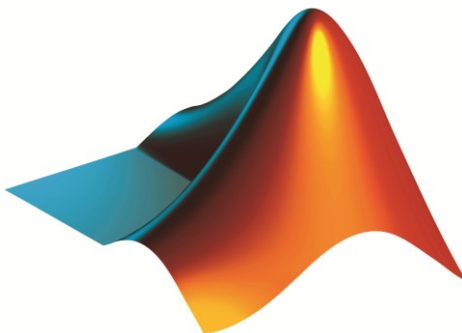
3 Použité programy

Pre vypracovanie mojej bakalárskej práce som použil nasledujúce programy:

- Matlab
- OpenSmile

3.1 Matlab

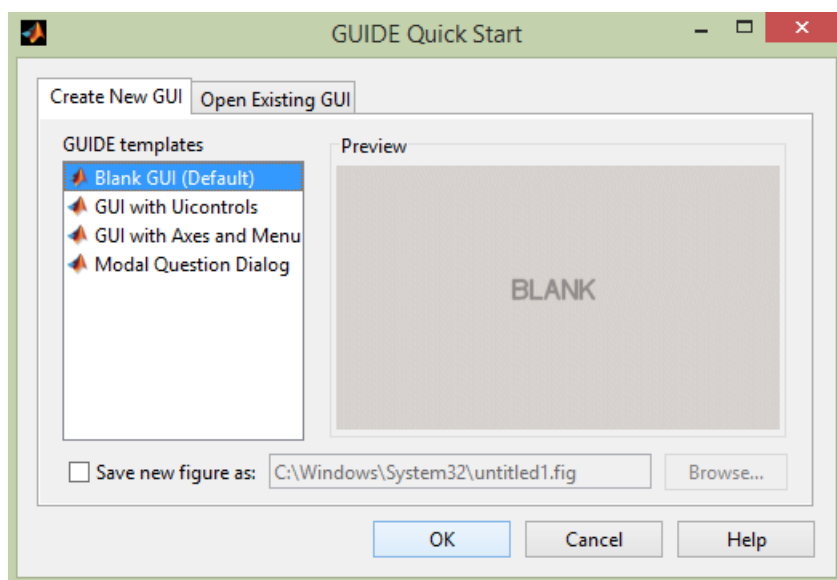
Matlab je integrovaným prostredím a pomocou neho máme veľa možností napr. matematické výpočty, analýzu a vizualizáciu dát, vývoj algoritmu, modelovanie, meranie a spracovanie dát, návrhy riadiacich a komunikačných systémov a mnoho ďalších výpočtov. Základom Matlabu je výpočtové jadro, prevádzajúce numerické operácie s maticami reálnych či komplexných čísiel. Teda Matlab je maticovo orientovaný.



Obrázok 3.1: *Logo Matlabu*

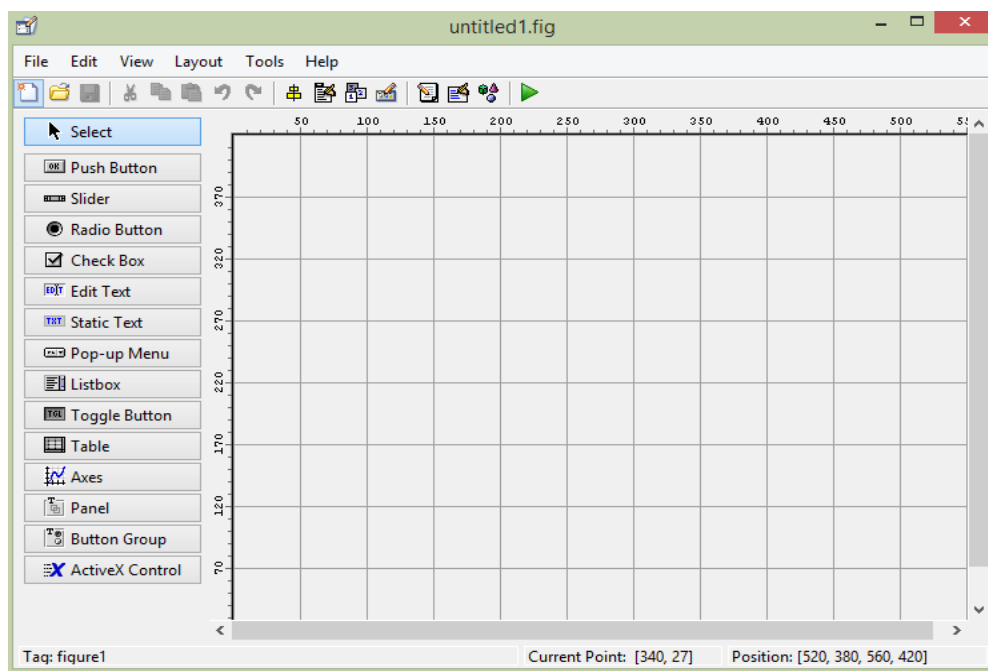
3.1.1 GUIDE - nástroj pre interaktívnu tvorbu grafického rozhrania

Vytváranie grafických rozhraní v prostredí Matlabu je pomerne jednoduché. Využíva sa na to najčastejšie **toolbox** s názvom Guide. To bude aj náš prípad. Po zadaní príkazu `guide` v **Command Window** sa otvorí okno s názvom **GUIDE Quick Start** obr.(3.2), ktoré nám ponúka niekoľko základných typov na vytvorenie grafického rozhrania (**Blank GUI**, **GUI with Uicontrols**, **GUI with Axes and Menu**, **Modal Question Dialog**), ako aj možnosť otvorenia už existujúceho GUI. [2]



Obrázok 3.2: *Guide Quick Start*

Pre vytvorenie nášho GUI si vyberieme možnosť **Blank GUI** (default). Sprievodca tvorbou GUI obr.(3.3) obsahuje všetky grafické objekty typu **unicontrol**, ktorými je možné ovládať aplikácie. Vzhľad vytvoreného GUI ukladá GUIDE do súboru s príponou *.fig a jeho zdrojový kód do súboru *.m.



Obrázok 3.3: *Vzhľad sprievodcu tvorbou GUI*

Vzhľad sprievodcu možno rozdeliť do troch skupín. Na ľavej strane sa nachádza tzv. **Component Palette**, na ktorej sú objekty **unicontrol**. Pracovnú plochu tvorí štvorcová sieť - **Layout Area**. Vodorovné ikony v hornej časti možno rozdeliť do dvoch skupín štandardné a špecifické pre GUI. Ikona v tvare zelenej trojuholníkovej šípky má funkciu tzv. **Active figure**. Uloží zmeny vzhľadu GUI do súboru obrázka *.fig a vytvorí (aj uloží) m- súbor s rovnakým názvom ako súbor s príponou *.fig. Je potrebné, aby oba súbory boli uložené v rovnakom priečinku.

3.2 OpenSmile

OpenSmile extrakcie príznakov je audio analytický nástroj, ktorý vám umožní extrahovať veľké audio (a v poslednej dobe aj video) funkcie priestorov postupne a rýchlo, a aplikovať metódy strojového učenia, klasifikovať a analyzovať dáta v reálnom čase. OpenSmile je pôvodne skratka pre reč a hudobnú interpretáciu, extrakcie rečových príznakov.

Opensmile je určený ako súbor nástrojov pre výskumníkov a vývojárov, ale niekedy môže byť používaný aj koncovým užívateľom. Je napísaný v jazyku C++ a je k dispozícii ako spustený súbor v príkazovom riadku, rovnako ako aj dynamická knižnica. Hlavné rysy OpenSmilu sú jeho schopnosti on-line prístupového spracovania a jeho modularita.

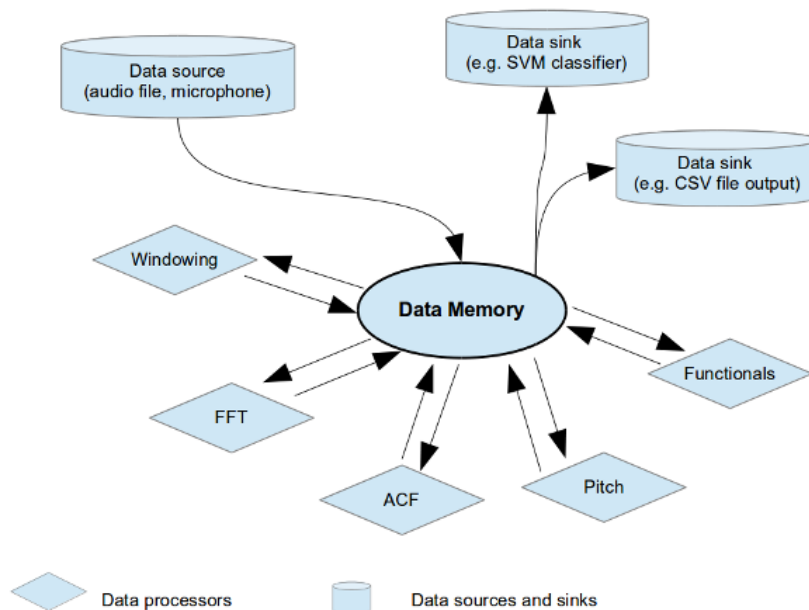
Hlavné vlastnosti sú:

- Je to cross-platform (Windows, Linux, Android, AOS).
- Ponúka aj inkrementálne a dávkové spracovanie.
- Účinne a veľmi rýchlo extrahuje veľké množstvo funkcií.
- Podporuje multi-threading pre paralelnú extrakciu príznakov.
- To je možné rozšíriť o nové vlastné komponenty a moduly plug-in.
- Podporuje zvukové súbory, ich vstupy a výstupy ako aj živý záznam zvuku a prehrávanie.
- Výpočet MFCC, PLP-CC, energie a delta regresné koeficienty a je plne kompatibilný s HTK.
- Má celý rád všeobecného spracovania audio signálu:
 - Fourierová transformácia
 - Kepstrum
 - Autolorelácia
- Generic a populárne formáty dát súborov sú podporované:
 - Hidden Markov Toolkit (HTK) súbory parametrov (čítanie / zápis)
 - WEKA ARFF súbory (v súčasnej dobe iba non-riedky) (čítanie / zápis)
 - Čiarkami oddelené hodnoty (CSV), text (čítanie / zápis)
 - LibSVM formáty funkcia file (zápis)

- Spracovanie reči v súvislosti s akustickými deskriptormi:
 - Energia signálu
 - MFCC a PLP-CC
 - Mel- / Bark- / Octave-scale spektra
 - Pitch (ACF a SHS algoritmy a Viterbi vyhladzovanie)
 - Kvalita hlasu (Jitter, Shimmer, HNR)
 - Linear Predictive Coding (LPC)
 - Linka Spektrálne pary (LSP)
 - Formanty
 - Spektrálnych deskriptorov tvaru (Roll-of, sklon, atď)

3.2.1 Architektúra OpenSmile

OpenSmile má veľmi modulárnu architektúru, určenú pre kumulatívne toky dát. Hlavnou súčasťou počítačov je **dataMemory**. Môže čítať viacero dát na raz. K dispozícii sú **data-source** komponenty, ktoré čítajú dáta z súborov alebo iných externých zdrojov a zoznámia ich s dataMemory. Potom sú tu komponenty spracovateľom dát, ktoré čítajú údaje a je potrebných ich uložiť a upraviť do novej vyrovnávacej pamäti - toto sú skutočné funkcie extrahovania. Nakoniec **data-sink** čítajú konečné údaje a uložia ich do súborov.

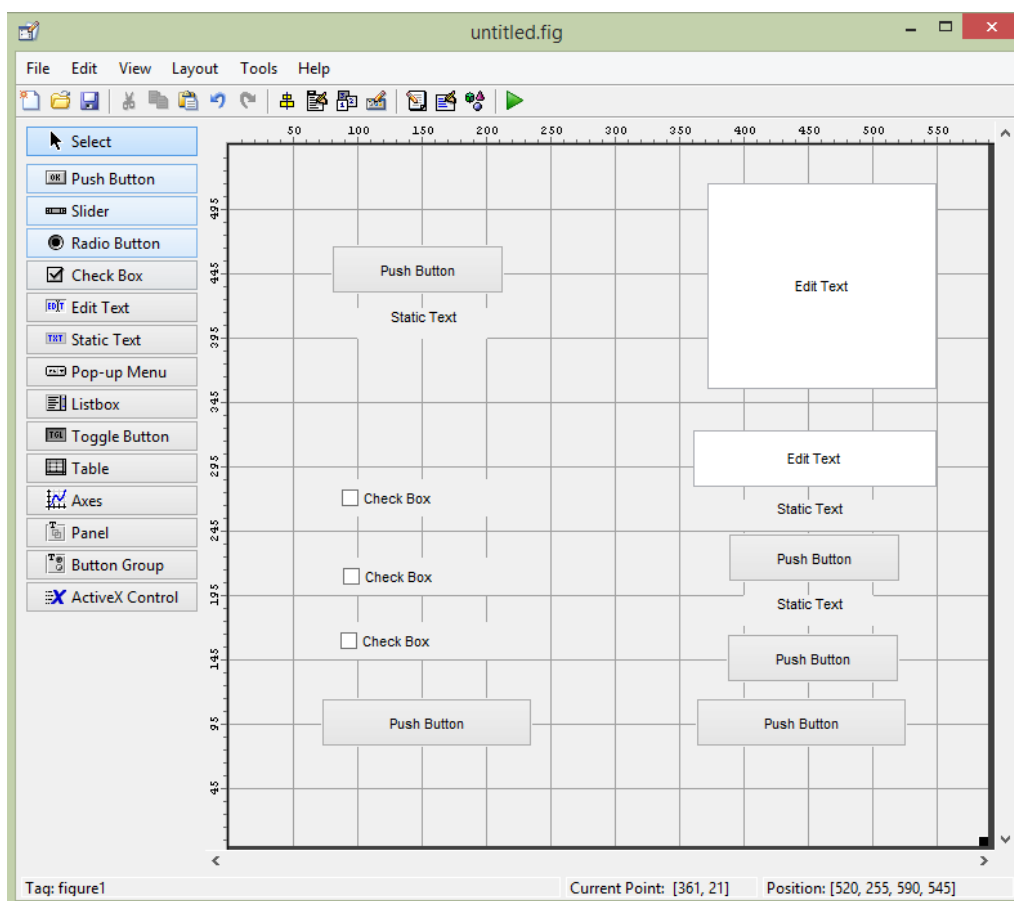


Obrázok 3.4: Architektúra OpenSmile

4 Návrh GUI pre extrakciu rečových príznakov

Samotný výpočet sa bude realizovať v súbore s príponou *.m. Pri tvorbe vizuálnej stránky GUI je potrebné poznať vstupy a výstupy. Vstupným súborom môjho GUI bude súbor wav. Môžeme si zvoliť výber jedného wav súboru až niekoľkých wav súborov, podľa potreby. Výstupom GUI bude extrahované wav do tabuľky. Tu máme dve možnosti otvorenia výstupu, prvou možnosťou otvorenia je výber v Microsoft Excel.

Druhou možnosťou výstupu, bude načítanie tabuliek extrakcie priamo do Matlabu. Vstup, výstup, štart, do GUI nám zabezpečí prvok Push Button. Analýzovanie prvkov prostredníctvom zvolených metód nám zabezpečí prvok check box. Popis spustených vykonávaných operácií, zadanie názvu výstupného súboru nám zabezpečí prvok edit text. Všetky použité prvky sa nachádzajú na ľavej lište. Ich výber je jednoduchý, postačí ich len umiestniť na pracovnú plochu a zoradiť podľa potreby. Ukážka návrhu dizajnu je na obr (4.1)



Obrázok 4.1: Prvotné usporiadanie prvkou

Ďalším krokom pri tvorbe GUI je úprava jednotlivých prvkov, ktoré sme si umiestnili na pracovnú plochu. Úpravu môžeme spraviť viacerými spôsobmi. Jednou možnosťou je použiť funkcie umiestnené na hornom paneli, druhou možnosťou je použiť tzv. **Inspector: unicontrol**, ktorý sa objaví po dvoj kliknutí na vybraný prvok, alebo po kliknutí na príslušnú ikonu v hornom paneli. Týmto spôsobom je možné upravovať všetky vlastnosti prvku (definovať pozíciu a veľkosť prvku, meniť farbu písma, prvku a pozadia, rovnako tak meniť názvy prvkov a ich **Tagov** a mnoho ďalšieho).

Každý prvok má pridelený **Tag**, tento "parameter" je pre ďalšie programovanie veľmi dôležitý. Môžeme ho považovať za tzv. ukazovateľ na daný prvok. Dobré zvolenými názvami **Tagov** si môžete zjednodušiť ďalšiu prácu s GUI. Zmena názvu **Tagu** je možné realizovať pomocou **Inspectora**.

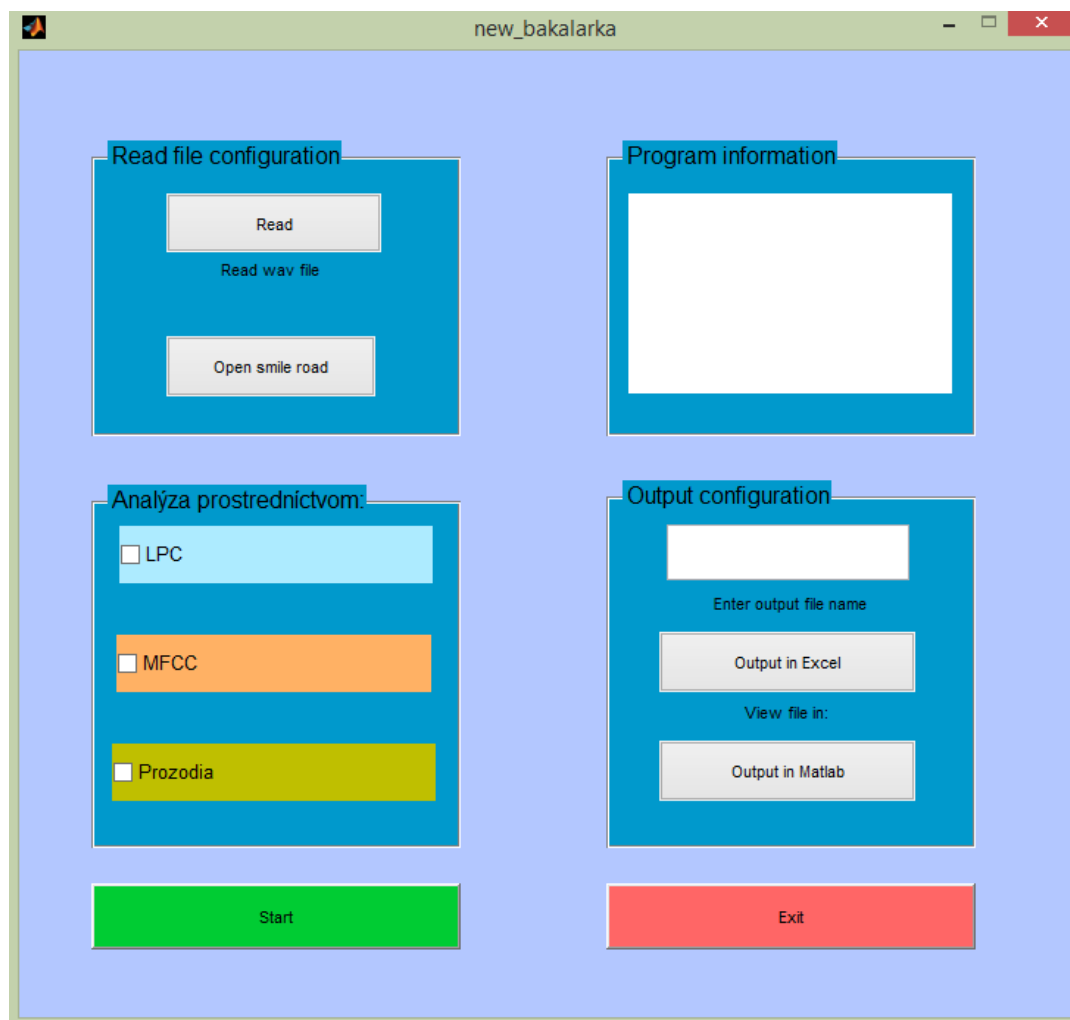
Po úprave jednotlivých prvkov sa nami vytvárané GUI zmenilo do podoby s ktorou bude pracovať užívateľ. Takéto GUI je zobrazené na obrázku (4.2).

K pôvodne vloženým prvkom pribudol ešte jeden s názvom Panel. V tomto prípade je prvok vložený len ako vizuálny prvok a neplní žiadnu funkciu. Názvy **Tagov** (ukazovateľov) pre jednotlivé prvky sú uvedené v Tabuľke (1.2). Tieto názvy budeme potrebovať pri písaní zdrojového kódu pre GUI.

Tabuľka 4.1: *Ukazovateľ (Tag) pre jednotlivé prvky*

TAG	Zobrazenie na ploche	TAG	Zobrazenie na ploche
Uipanel R.	V ľavo hore	Uipanel P.	V pravo hore
Push button R.	V ľavo v Uipanel R.	Edit text	V Uipanel P.
Static Text	Pod Push button R.	Uipanel O.	V pravo dole
Uipanel A.	V ľavo dole	Edit text	V Uipanel O.
Check box LPC	V Uipanel A.	Static text	V Uipanel O.
Check box MFCC	V Uipanel A.	Push button O.	V Uipanel O.
Check box P.	V Uipanel A.	Static text	V Uipanel O.
Push button Š	Dole	Push button O.	V Uipanel O.
Push button S	Dole	Push button R	V Uipanel R

Jednotlivé prvky sú v tabuľke zobrazené pomocou skratiek, písmeno za názvom **Tagu** znamená začiatkové písmeno ďalšieho pomenovania prvku. GUI obsahuje 4x Uipanel, 3x Static text, 3x Check box, 2x Edit text a 6x Push button.



Obrázok 4.2: GUI upravené do konečnej podoby

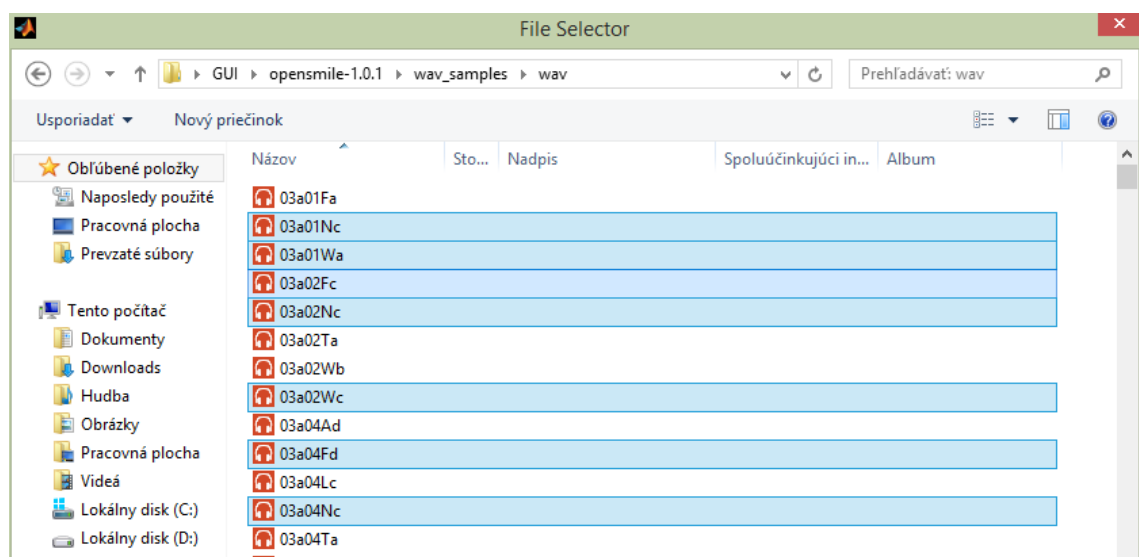
V takejto podobe môžeme GUI prvýkrát spustiť, využívame na to funkciu s horného panela s názvom **Run Figure** (zelená šípka). Táto funkcia zároveň vytvorí a uloží m-súbor s rovnakým názvom ako má náš súbor s príponou *.fig. Je nutné poznamenať, že názvy súborov musia byť uložené bez diakritiky a medzier v názve.

Vytvorený m-súbor obsahuje pred vytvorené funkcie prislúchajúce k jednotlivým prvkom. Do funkcií je nutné vpísať kód, ktorý sfunkční GUI. Najdôležitejšiu funkciu bude plniť tlačidlo Štart, ktoré bude obsahovať väčšinu kódu.

Tlačidlo Read

Prostredníctvom tohto tlačidla máme možnosť si vyberať koľko wav súborov sa nám bude spracovávať. Na výber máme jeden až viac vstupných súborov. Bez načítania vstupných súborov sa nám nevykoná potrebná analýza. Všetko čo sa bude nachádzať v tele funkcie tlačidla sa nám vykoná po stlačení. Potrebné je doplniť, že cestu k wav súborom je treba kliknúť a sú uložené v priečinku wav_samples.

```
function Read_Callback(hObject, eventdata, handles)
% nacitanie a vyber wavka
[FileName,PathName,FilterIndex] = uigetfile({'*.wav'},'File
Selector','MultiSelect','on');
% dátava štruktúra
data = struct('path', PathName, 'file', FileName(:, :));
ulozenie_predošlej_štrukturi
set(handles.Read, 'UserData', data);
```



Obrázok 4.3: *Priečinnok s wav súbormi*

Tlačidlo OpenSmile road

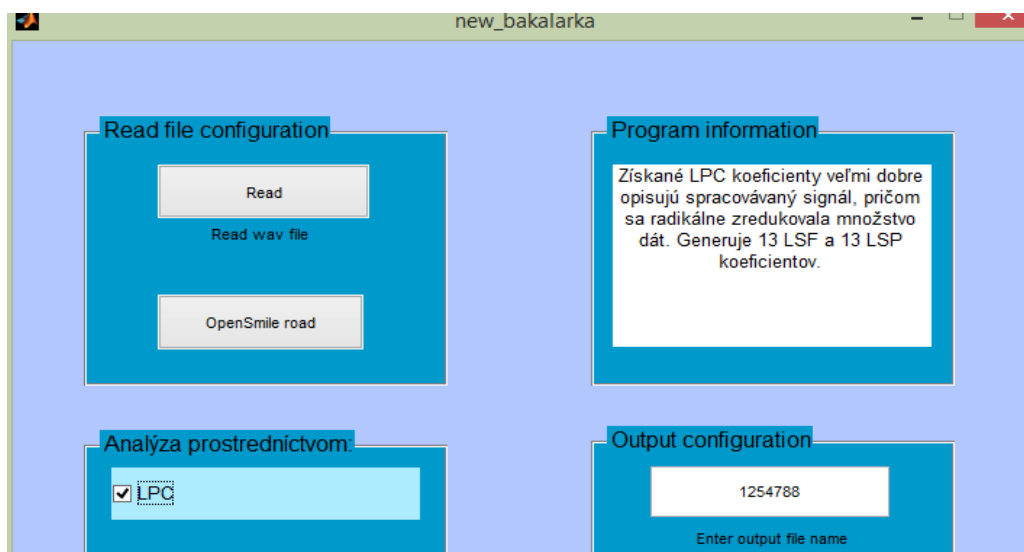
Toto tlačidlo slúži na načítanie cesty k Opensmilu z toho dôvodu, pretože používame poskladaný textový reťazec, aby sme spustili program. Keď bude používať aplikáciu iný používateľ na svojom PC, tak súbor OpenSmile môže mať uložený na konkrétnom mieste v PC. Týmto tlačidlom nájdeme OpenSmile a načítame cestu k nemu.

```
function road_Callback(hObject, eventdata, handles)
folder_name = uigetdir;
data = struct('path', folder_name);
% Nastavenie cesty
set(handles.road, 'UserData', data);
```


Checkbox LPC, MFCC, Prozodia

Tieto check boxy nám analyzujú vybranú metódu, ktorú si užívateľ zvolí. Defaultne je nastavené, že užívateľ si môže zvoliť iba jednu z týchto metód, aby nedošlo ku chybe. Každá jedna analýza má svoj informačnú časť, tá popisuje čo sa robí a je zobrazená v pravo hore v GUI.

```
function LPC_Callback(hObject, eventdata, handles)
set(handles.MFCC, 'Value', 0);
set(handles.Prozodia, 'Value', 0);
% Premenná, ktorá sa volá pri volaní textu
a = 'Získané LPC koeficienty veľmi dobre opisujú spracovávaný  
signál, pričom sa radikálne zredukovala množstvo dát. Generuje  
13LSF a 13LSP koeficientov. '
set(handles.text1, 'String', a)
```

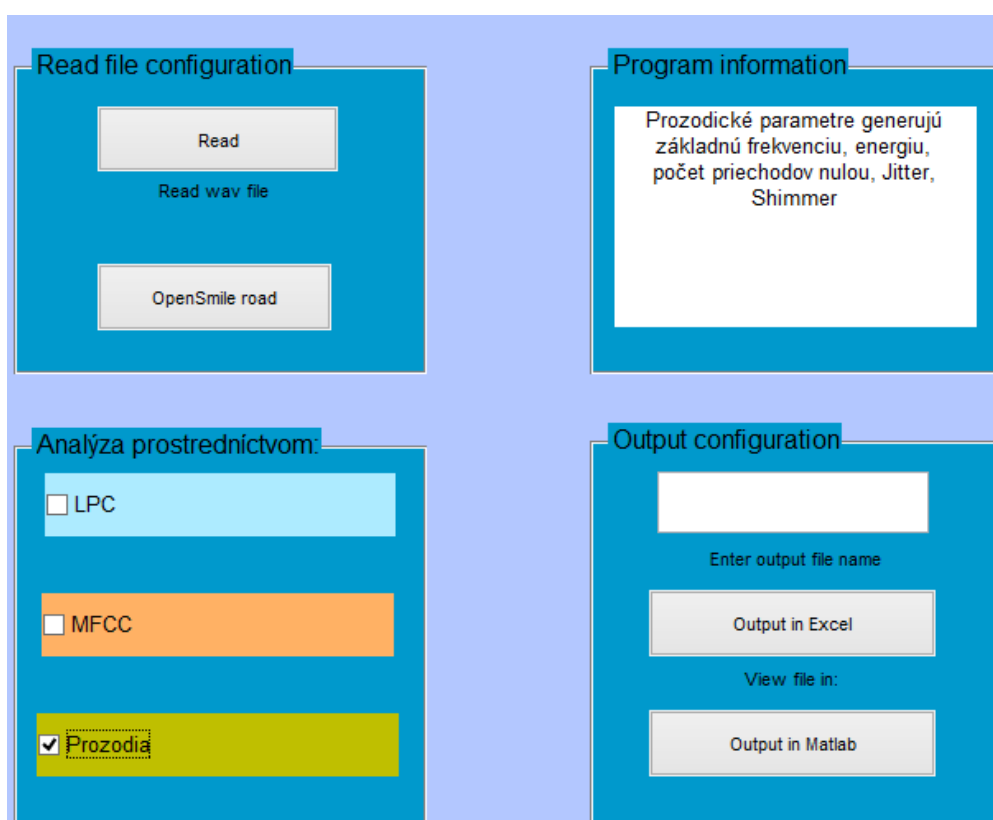


Obrázok 4.4: Check box LPC

```
set(handles.LPC, 'Value', 0);
set(handles.Prozodia, 'Value', 0);
mfcc = 'Generuje 13 mfcc parentrov 13 delta mfcc parametrov a 13  
delta delta mfcc parametrov. '
set(handles.text1, 'String', mfcc)
```



Obrázok 4.5: Check box MFCC



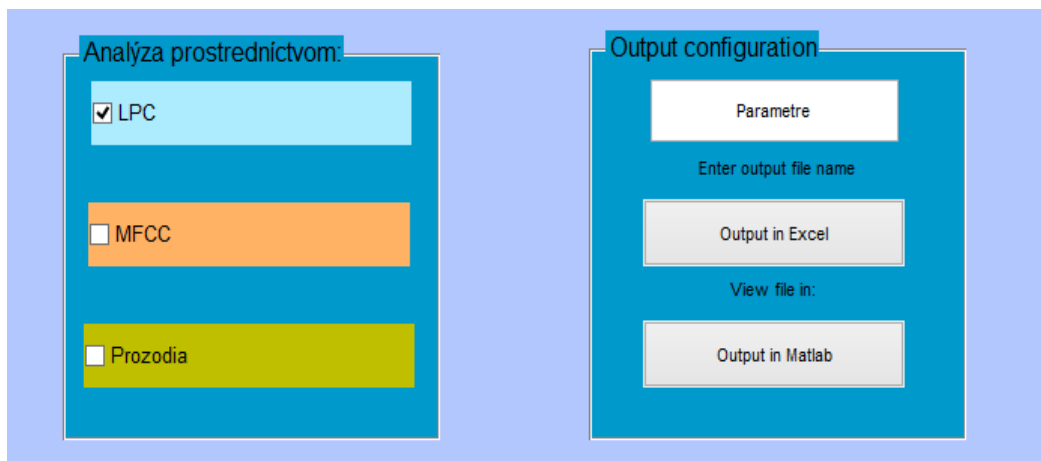
Obrázok 4.6: Check box Prozodia

```
set(handles.MFCC,'Value',0);
set(handles.LPC,'Value',0);
proz = 'Prozodické parametre generujú základnú frekvenciu,
energiu, počet priechodov nulou, Jitter, Shimmer '
set(handles.text1,'String',proz)
```

Edit text - názov výstupného súboru

Prostredníctvom tohto editu si môžeme zvoliť, ako sa bude náš výstupný súbor menovať. Výstupný súbor si potom nájdeme v Output configurátor obr(4.7). Cesta k súboru je uložená v štart tlačídku.

```
output_name = get(handles.Output_name,'String');
```



Obrázok 4.7: Output configurátor

Output in Excel, Matlab

Výstupný súbor nám zobrazí v programe Excel. Prvým krokom bude, že klikneme na tlačídko Output in Excel obr(4.7) a dostaneme sa prekliknutím do OpenSmile v zložke wav_samples a načítame si výstupný súbor.

Výstupný súbor sa nám pomocou príkazov upraví, tým že sa oddelia prvé dva stĺpce. Následne sa zmení aj názov súboru, ktorý sa otvorí v programe Excel obr(4.8). Druhý krok je zobrazením a načítaním do Matlabu prostredníctvom tlačidla Output in Matlab.

```
function Vystup_Callback(hObject, eventdata, handles)
% Výber súboru
[FileName,PathName,FilterIndex] = uigetfile({'*.csv'},'File
Selector');
lpc_matlab=importdata(fullfile(PathName,FileName),';',1);
% Oddelenie riadkov a stĺpcov
W_lpc=lpc_matlab.data(:,3:end);
[pathstr,name,ext] = fileparts(FileName);
new_filename = fullfile(PathName,[name ,'_file2', '.xlsx']);
data = struct('path', new_filename);
set(handles.Vystup,'UserData',data);
```

C	D	E	F	G	H	I	J	K	L	M	N
mfcc[0]	mfcc[1]	mfcc[2]	mfcc[3]	mfcc[4]	mfcc[5]	mfcc[6]	mfcc[7]	mfcc[8]	mfcc[9]	mfcc[10]	mfcc[11]
-9.800679e	-6.351812e	6.366284e	-3.953387e	1.631473e	-8.934741e	-1.127378e	1.432028e	-2.511459e	-8.912513e	5.736523e	1.889027e
-9.398537e	-6.506074e	7.515741e	1.082421e	1.924307e	2.973140e	1.427936e	-5.996706e	6.902297e	-4.324023e	2.298503e	-5.473728e
-8.414536e	-2.902661e	2.647817e	1.470352e	3.801589e	2.893675e	2.322073e	-5.859485e	4.280703e	6.481034e	-2.816941e	3.563301e
-8.366991e	-2.034868e	2.005057e	8.656265e	6.229575e	5.715685e	1.569435e	-2.038077e	6.298519e	5.800853e	6.099584e	6.878737e
-8.488757e	-5.279270e	2.941830e	8.957716e	1.600233e	1.558561e	1.382494e	-6.683702e	4.496025e	1.319896e	-1.589868e	9.039298e
-8.437041e	-1.352554e	4.130918e	5.665537e	9.447659e	1.466382e	1.516894e	-4.872468e	-7.052212e	8.218401e	-4.326279e	8.413538e
-8.750195e	-4.999008e	2.943110e	1.274968e	4.548991e	1.461999e	2.313260e	5.399370e	1.509356e	1.967947e	5.671735e	4.323311e
-8.769511e	1.699474e	9.089090e	-1.714191e	7.486926e	5.369562e	7.675499e	2.248807e	-2.068689e	1.522707e	-8.662920e	1.083941e
-8.893689e	-2.112709e	6.289835e	2.811045e	7.668191e	7.987019e	1.259075e	-5.494277e	1.272220e	1.265267e	-8.460892e	-8.838926e
-8.886122e	1.117859e	4.591930e	1.305687e	1.007349e	1.240433e	1.964888e	-3.965058e	-1.117247e	1.893740e	-1.094653e	1.250063e
-9.363448e	-4.440295e	7.229089e	8.414612e	1.842667e	9.881883e	2.861751e	-6.955390e	1.117077e	1.743684e	-1.555967e	3.382248e
-9.429539e	-2.045344e	9.469287e	1.269652e	-4.665848e	1.254953e	1.693777e	-5.131162e	3.546204e	5.733306e	-5.672104e	3.746833e
-9.480289e	-2.246263e	9.072435e	2.494273e	4.858480e	3.668502e	2.850269e	6.393850e	4.281547e	1.037545e	-1.011821e	1.401308e
-9.451853e	-1.788136e	1.005147e	4.530710e	1.925783e	-1.697857e	2.011545e	-1.704278e	6.525216e	9.731544e	-6.867296e	4.795394e
-9.564961e	-3.662683e	7.099742e	2.705677e	3.903600e	6.797091e	1.627043e	-5.543932e	8.017865e	8.604316e	1.315892e	-3.180191e
-9.629916e	-5.164676e	5.013738e	9.787153e	-9.222178e	2.051272e	1.597380e	-8.392069e	2.307883e	1.237199e	2.767590e	-1.799391e
-9.729118e	-4.848549e	6.700142e	2.407665e	3.359298e	4.514850e	9.875542e	1.946764e	-3.924894e	1.108601e	-3.957237e	4.369117e

Obrázok 4.8: Output in Excel

Tlačidlo Start

Prostredníctvom tohto tlačidla sa nám spúšťa celá výpočtová operácia, ktorú vykonáva OpenSmile spolu s Matlabským GUI. OpenSmile sa spúšťa pomocou dlhého textového reťazca, ktorý obsahuje cestu k daným súborom. Reťazec má nasledujúcu podobu:

```
'C:\Users\Jakub123\Desktop\opensmile-1.0.1\SMILEExtract -C  
C:\Users\Jakub123\Desktop\opensmile-1.0.1\config/lpc_matlab.conf  
-I  
C:\Users\Jakub123\Desktop\opensmile  
1.0.1\wav_samples/speech01.wav -O  
C:\Users\Jakub123\Desktop\opensmile-1.0.1\vystup'
```

Z tohto reťazca bolo potreba poskladať cestu tak, aby sme si mohli vybrať všetko čo potrebujeme pre extrakciu a následne vybrať výsledný spracovaný súbor.

```
function Start_Callback(hObject, eventdata, handles)  
  
opensmile_path_data = get (handles.road, 'UserData');  
smileextract = fullfile(opensmile_path_data.path, 'SMILEExtract -C  
' );  
lpc_config = fullfile(opensmile_path_data.path,  
'config/lpc_matlab.conf -I ');  
mfcc_config = fullfile(opensmile_path_data.path,  
'config/mfcc_matlab.conf -I ');  
Prozodia_config = fullfile(opensmile_path_data.path,  
'config/priznaky_matlab.conf -I ');  
% Prevzatie nacistanych wav suborov z read tlacidka  
data = get (handles.Read, 'UserData');  
zadefinovanie nazvu vystupneho suboru  
output_name = get (handles.Output_name, 'String');  
%cesta k vystupnemu suboru  
output_file = fullfile(data(1,1).path, [output_name, '.csv']);
```

Ďalšia časť kódu obsahuje výber možnosti pre určenie, ktorú metódu si vyberieme na extrakciu (LPC, MFCC, Prozodia):

```
if (get(handles.LPC, 'Value'))  
for m=1:length(data)  
lpc_matlab=[smileextract, lpc_config,  
fullfile(data(1,m).path,data(1,m).file), ' -O ', output_file];  
status=dos(lpc_matlab);
```

```

end

elseif (get(handles.MFCC, 'Value'));

for r m=1:length(data)

mfcc_matlab=([smilextract, mfcc_config,
fullfile(data(1,m).path,data(1,m).file),' -O ', output_file]);

status=dos(mfcc_matlab);

end

elseif (get(handles.Prozodia, 'Value'));

for m=1:length(data)

Prozodia_matlab=([smilextract, Prozodia_config,
fullfile(data(1,m).path,data(1,m).file),' -O ', output_file]);

status=dos(Prozodia_matlab);

end

```



Obrázok 4.9: Plne funkčné GUI

Záver

Cieľom tejto bakalárskej bolo navrhnuť GUI v programe Matlab pre extrakciu rečových príznakov, pre uľahčenie výpočtov v programe OpenSmile. K tomuto účelu som využil nástroj pre interaktívnu tvorbu grafického rozhrania (GUIDE), ktoré disponuje handles grafik pre uľahčenie tvorby GUI. Najprv som v práci popísal vznik a vlastnosti reči. V druhej kapitole som popísal parametrizáciu rečového signálu, ktorá bola potrebná pre vysvetlenie pedspracovania signálu pred jeho samotným spracovaním. V tretej kapitole som popísal použitý software Matlab a OpenSmile.

Čo sa týka samotného návrhu GUI, tak sa podarilo všetko navrhnuť podľa predstáv. Podmienkou mojej práce bolo, aby sa mohlo v programe spracovávať viacej/viacero súborov naraz a pri spracovaní súboru si zvoliť danú metódu spracovania a následne zadať názov výstupného súboru. Na konci práce bolo nutné ešte previesť výber programu OpenSmile, čo znamenalo, že keby užívateľ prenášal navrhnuté GUI na iné PC, tak by to nemalo svoju funkčnosť z dôvodu uloženia programu. Najväčší problém pri návrhu sa vyskytol s prepájaním reťazca pri ukladaní výstupného súboru, kde súbor bolo najprv treba odstrániť riadky a stĺpce a potom uložiť ako nový súbor. Všetko sa nakoniec úspešne podarilo.

Táto bakalárska práca mi priniesla poznatky v grafickom prostredí programu Matlab, overenie zručnosti programovania a poznatky o komunikácii človeka s počítačom.

Použitá literatura

- [1] ZAPLATÍLEK, Karel a Bohuslav DOŇAR. *MATLAB pro začátečníky*. 1. vyd. Praha: BEN - technická literatura, 2003. ISBN 80-730-0095-4.
- [2] ZAPLATÍLEK, Karel a Bohuslav DOŇAR. *MATLAB: tvorba uživatelských aplikací*. 1. vyd. Praha: BEN - technická literatura, 2004, 209 s. ISBN 80-730-0133-0.
- [3] PSUTKA, Josef, Luděk MULLER, Jindřich MATOUŠEK a Vlasta RADOVÁ. *Mluvíme s počítačem česky*. Vyd. 1. Praha: Academia, 2006, 746 s. ISBN 80-200-1309-1.
- [4] EYBEN, Florian, Felix WENIGER, Martin WÖLMER a Björn SCHULLER. *OpenSmile*. München: Institute for Human-Machine Communication Technische Universitaet Muenchen (TUM), 2013, 169 s.

Zoznam príloh

Súčasti BP/DP je CD/DVD.

Adresárová štruktúra priloženého CD/DVD:

Spúšťačí súbor programu

Program OpenSmile

wav súbory